OPPORTUNITY TRACKING INFORMATION SYSTEM

INVENTORS:

J Dale Debber
Dan Ribb
Rodney Bennet

Prepared by:

Greg T. Sueoka
Reg. No. 33,800
Fenwick & West LLP
Two Palo Alto Square
Palo Alto, CA 94306

Express Mailing Label # EL482470653US

1 <u>OPPORTUNITY TRACKING INFORMATION SYSTEM</u>

2

3 Inventors: J Dale Debber; Dan Ribb; and Rodney Bennet

4

5 <u>Cross-Reference to Related Application</u>

6 This application claims priority under 35 U.S.C § 119(e) from co-pending U.S.

7 Provisional Application No. 60/178,168, filed on January 26, 2000 by J Dale Debber, *et*

8 *al.*, entitled "Opportunity Tracking Information System," the subject matter of which is

9 fully incorporated herein by reference.

10

11 <u>Field of the Invention</u>

12 The present invention relates generally to communication network systems, and more

13 particularly to a system, method and computer medium, for identifying, tracking and

14 managing the completion of tasks to achieve improved process-flow performance.

15

16 <u>Background of the Invention</u>

17 The complexity, size and distribution of many organizations (including businesses) are

18 ever increasing and changing. The rapid growth in size and complexity of an organization

19 typically leads to a large number of tasks (i.e., processes) that must be accomplished in a

20 timely manner for any organization to be successful. The tasks may vary in scope based upon

21 the context of the situation. For example, the tasks may include routine administrative

22 matters, customer service requests, specific projects, manufacturing steps, and business and

23 other related service-type processes. A common aspect amongst these tasks is that they are

1   performance-driven, frequently measured by the timely completion of milestones along with

2   the appropriate level of quality assurance. Conventional techniques have failed to effectively

3   address the change in dynamics of an organization, whereupon the increasing complexity and

4   quantity of tasks undertaken require effective and efficient management integrated seamlessly

5   into the process flow.

6   Additionally, there is a growing trend where organizations are increasing and

7   dispersing their workforce in the market place to enable them to be closer to their customers;

8   however, the organizations are simultaneously facing significant labor shortages. The

9   difficulty in finding qualified personnel to fill various positions within an organization

10 frequently leads to an accommodation, wherein an individual (e.g., employee) is assigned the

11 responsibilities of multiple positions. For example, one drawback associated with the situation

12 of having an employee perform more than one role within a business is that it leads to high

13 employee dissatisfaction and turnover. This, in turn, results in workflow interruption and

14 inefficiency when tasks and processes must be reassigned to another employee. Often times

15 the turnover in employees can lead to tasks and processes being unattended and remaining

16 uncompleted. This situation is frequently problematic for the success and prosperity of any

17 growing organization.

18   Accordingly, it is desirable to provide a method and system that automatically

19 addresses the above-mentioned difficulties of managing process-flow performance in a

20 complex and widely situated organization, prone to increases in work-force size. It is further

21 desirable to provide a method and system that identifies and tracks workflow and work

22 processes so that they may be managed efficiently and through to completion. It is further

23 desirable to provide a method and system that reduces the effect of disruption to workflow

1   processes caused by individuals changing positions regularly, so as to maintain continuity of

2   action and purpose in particular tasks at hand. It is further desirable to provide a method and

3   system for tracking workflow in the manner that can be easily accessed and consistently

4   distributed amongst the strata of members within an organization.

5

6   Summary of the Invention

7   The opportunity tracking information system present invention includes a method

8   for managing tasks in a computer communication network. The method includes

9   accessing a centralized database, containing information on tasks and their status. The

10 status of each task includes, but is not limited to: a description of the task, and the person

11 (or persons) within the organization responsible for completing the task. The program

12 used to access the server is preferably a browser or other computer program designed for

13 access across a network. After identifying the person using the program as the "user",

14 the status of all tasks associated with that user or individuals for whom the user has

15 authority to monitor is displayed in a formatted manner. The user may then update the

16 status of tasks that he is permitted to view. In general, a user may view the status of a

17 task if the user is responsible for accomplishing the task or if the person responsible for

18 accomplishing the task reports to the user within the organizational structure. Thus, the

19 ability to view tasks parallels an organizational chart since a user at a position in the

20 organizational chart can view tasks for that position and tasks for any position in the

21 organizational chart below that position. Among the updates that the user may

22 accomplish are creating new tasks for himself or for those who report to him, marking

-3-

1   tasks as being finished, and assigning (moving) a task to other individuals or positions

2   within the organization.

3   More specifically in an embodiment using the Internet, the method of the present

4   invention includes the step of accessing a first server from at least one client program such as a

5   browser. The first server accesses an online account to retrieve status information associated

6   with the tasks stored on a database for display currently to the client browser. In response to

7   receiving a user instruction for managing the tasks, updates of the status information are

8   generated. The updated status information is provided from the server to the client browser

9   for display thereon.

10   With the above method, the present invention is designed to track anomalies, routine

11   automatic tasks, and manually inputted tasks by the positions of those individuals as defined

12   within an organization. This hierarchical approach facilitates management and administration

13   of projects. By providing a management tool that in one embodiment is implemented over a

14   network, individuals may uniformly identify, track and manage the completion of tasks.

15   Furthermore, administrators may implement the opportunity tracking information system of

16   the present invention in a scalable manner that is useful to those organizations having a work

17   force widely situated throughout multiple locations.

18   In one key aspect of the present invention, the opportunity tracking information

19   system merges the organizational structure of an organization inherently into the processes for

20   creating, editing, moving, adding, removing or viewing tasks. In particular, the opportunity

21   tracking information system provides three general types of users: users, managers and

22   administrators. A user refers to a position in the organization chart that does not have any

23   nodes or positions below it. Thus, the system provides a user typically with only the ability to

-4-

1 view tasks for a particular position or person; and can create, edit, add or remove tasks only

2 for that position. A manager refers to a position in the organization chart that has at least one

3 node or position below it. A manager has the same rights as a user but can also perform any

4 of the actions on tasks for the nodes below (or reporting to) the manager position. The system

5 therefore provides managers with the ability to view tasks for their position and any position

6 below or reporting to their position; and the ability to create, edit, add or remove tasks their

7 position and any position below or reporting to their position. An administrator refers to a

8 position that is responsible for ensuring the system and the processes for tasks that the system

9 allows match the organization chart of an organization. The administrator has the same rights

10 as a user but can also modify control parameters for operation of the system. The

11 administrators have the capability to change, which positions report to other positions, and

12 thus, the tasks that each position may view, edit, modify or assign. It should be understood

13 that a particular person could be both a manager and an administrator type and thus the system

14 would allow that user to perform actions of either type. Accordingly ever user will be able to

15 perform different actions and view different data depending the type of user they are.

16   Another aspect of the present invention includes a management tool that allows

17 multiple views of what is to be done within an organization. Individuals are allowed to view

18 information about the tasks to be accomplished, to update the status of the work items in an

19 interactive manner and to enter new work items. The extent to which an individual may

20 manipulate the attributes of a work item is controlled by the individual's class and position

21 within an organization.

22

Brief Description of the Drawings

FIG. 1 is block diagram illustrating a first embodiment of the opportunity tracking information system for the automatic management of process flow performance in accordance with the present invention.

FIG. 2 is a flowchart that illustrates a process by which a user gains access to the system of FIG. 1 in accordance with the present invention.

FIG. 3 is a flowchart illustrating a method for determining whether a user has previously established a customized home page once the user has successfully logged into the system.

FIG. 4 is a flowchart illustrating a method of determining how status information (e.g., tasks and anomalies) is reviewed or edited.

FIG. 5 is a flowchart illustrating a method for providing status information concerning the administrative features in accordance with the present invention.

FIG. 6 illustrates an exemplary graphical representation of a user interface for allowing an individual to input the user identification number (User ID) and password.

FIG. 7 illustrates an exemplary graphical representation of a user interface having a default home page combined with the status information retrieved from the database.

FIG. 8 illustrates an exemplary graphical representation of a user interface enabling a user to describe in detail the anomaly and the anomaly type in accordance with the present invention.

FIG. 9 illustrates an exemplary graphical representation of a user interface enabling a manager to assign a priority level to the anomaly in accordance with the present invention.

1   FIG. 10 illustrates an exemplary graphical representation of a user interface enabling

2   the completion and approval of an anomaly in accordance with the present invention.

3   FIG. 11 illustrates an exemplary graphical representation of a user interface (review

4   screen) enabling a manager to review any anomaly.

5   FIG. 12 illustrates an exemplary graphical representation of a user interface enabling a

6   user to review pending tasks and anomalies.

7   FIG. 13 illustrates an exemplary graphical representation of a user interface that allows

8   the user to enter the information about a task.

9   FIG. 14 illustrates an exemplary graphical representation of a user interface enabling

10   the input of information used to track the date that a task was inputted and the priority

11   assigned thereto.

12   FIG. 15 illustrates an exemplary graphical representation of a user interface enabling a

13   manager to assign the amount of time for a task to be completed in accordance with the

14   present invention.

15   FIG. 16 illustrates an exemplary graphical representation of a user interface where

16   managers are allowed to check when they want to receive email notification.

17   FIG. 17 illustrates an exemplary graphical representation of a user interface indicating

18   the occurrence of a task on the input screen in accordance with the present invention.

19   FIG. 18 illustrates an exemplary graphical representation of a dialog box that allows

20   the user to assign the occurrence of the task.

21   FIG. 19 illustrates an exemplary graphical representation of a user interface for

22   reviewing tasks.

1   FIG. 20 illustrates an exemplary graphical representation of a user interface for

2   escalating a task that remains incomplete.

3   FIG. 21 illustrates an exemplary graphical representation of a user interface that

4   indicates position classes and positions in accordance with the present invention.

5   FIG. 22 illustrates an exemplary graphical representation of a user interface enabling a

6   manager to review all the templates that exist for projects assigned thereto.

7   FIG. 23 illustrates an exemplary graphical representation of a user interface for

8   reviewing status information according to the calendar option in accordance with the present

9   invention.

10   FIG. 24 illustrates an exemplary graphical representation of a user interface for

11   reviewing anomalies and tasks on a weekly basis.

12   FIG. 25 illustrates an exemplary graphical representation of a user interface enabling

13   administrators to enter updates to the status information.

14   FIG. 26 illustrates an exemplary graphical representation of a user interface enabling

15   disciplines to be assigned to a project.

16   FIG. 27 illustrates an exemplary graphical representation of a user interface for

17   indicating project bugs in accordance with the present invention.

18   FIG. 28 illustrates an exemplary graphical representation of a user interface listing all

19   of the anomaly types that are maintained and the new ones that are added in accordance with

20   the present invention.

21   FIG. 29 is a block diagram of a first embodiment for the client computer.

22   FIG. 30 is a block diagram of the memory unit of FIG. 29.

1   FIG. 31 is a block diagram of an organizational chart with the class and position for

2   different individuals as used by the system of the present invention.

3   FIG. 32 is a flowchart of one embodiment for the operation of the OTIS system

4   demonstrating its integration of organizational structure with the work process flow.

5

6   Detailed Description of the Embodiments

7   A system and method for displaying, assigning, modifying, deleting, creating and

8   tracking tasks in a computer system is described. In the following description, for purposes of

9   explanation, numerous specific details are set forth in order to provide a thorough

10   understanding of the invention. It will be apparent, however, to one skilled in the art that the

11   invention can be practiced without these specific details. In other instances, structures and

12   devices are shown in block diagram form in order to avoid obscuring the invention.

13   Reference in the specification to "one embodiment" or "an embodiment" means that a

14   particular feature, structure, or characteristic described in connection with the embodiment is

15   included in at least one embodiment of the invention. The appearances of the phrase "in one

16   embodiment" in various places in the specification are not necessarily all referring to the same

17   embodiment.

18   Some portions of the detailed description that follows are presented in terms of

19   algorithms and symbolic representations of operations on data bits within a computer memory.

20   These algorithmic descriptions and representations are the means used by those skilled in the

21   data processing arts to most effectively convey the substance of their work to others skilled in

22   the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of

23   steps leading to a desired result. The steps are those requiring physical manipulations of

1   physical quantities. Usually, though not necessarily, these quantities take the form of

2   electrical or magnetic signals capable of being stored, transferred, combined, compared, and

3   otherwise manipulated. It has proven convenient at times, principally for reasons of common

4   usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers,

5   or the like.

6   It should be borne in mind, however, that all of these and similar terms are to be

7   associated with the appropriate physical quantities and are merely convenient labels applied to

8   these quantities. Unless specifically stated otherwise as apparent from the following

9   discussion, it is appreciated that throughout the description, discussions utilizing terms such as

10   "processing" or "computing" or "calculating" or "determining" or "displaying" or the like,

11   refer to the action and processes of a computer system, or similar electronic computing device,

12   that manipulates and transforms data represented as physical (electronic) quantities within the

13   computer system's registers and memories into other data similarly represented as physical

14   quantities within the computer system memories or registers or other such information

15   storage, transmission or display devices.

16   The present invention also relates to an apparatus for performing the operations herein.

17   This apparatus may be specially constructed for the required purposes, or it may comprise a

18   general-purpose computer selectively activated or reconfigured by a computer program stored

19   in the computer. Such a computer program may be stored in a computer readable storage

20   medium, such as, but is not limited to, any type of disk including floppy disks, optical disks,

21   CD-ROMs, and magnetic-optical disks, read-only memories (ROMs), random access

22   memories (RAMs), EPROMs, EEPROMs, magnetic or optical cards, or any type of media

23   suitable for storing electronic instructions, and each coupled to a computer system bus.

-10-

1   Furthermore, the computers referred to in the specification may include a single process or

2   may be architectures employing multiple processor designs for increased computing

3   capability.

4   The algorithms and displays presented herein are not inherently related to any

5   particular computer or other apparatus. Various general-purpose systems may be used with

6   programs in accordance with the teachings herein, or it may prove convenient to construct

7   more specialized apparatus to perform the required method steps. The required structure for a

8   variety of these systems will appear from the description below. In addition, the present

9   invention is not described with reference to any particular programming language. It will be

10 appreciated that a variety of programming languages may be used to implement the teachings

11 of the invention as described herein, and any references below to specific languages are

12 provided for disclosure of enablement and best mode of the present invention.

13   Moreover, the present invention is claimed below as operating on or working in

14 conjunction with an information system. Such an information system as claimed may be the

15 entire opportunity tracking information system as detailed below in the preferred embodiment

16 or only portions of such a system. For example, the present invention can operate with an

17 information system that need only be a browser in the simplest sense to present and display

18 objects. Thus, the present invention is capable of operating with any information system from

19 those with minimal functionality to those providing all the functionality disclosed herein.

20   Reference will now be made in detail to several described embodiments of the present

21 invention, examples of which are illustrated in the accompanying drawings. Wherever

22 practicable, the same reference numbers will be used throughout the drawings to refer to the

23 same or like parts.

1   Referring now to FIG. 1, there is shown an example of a system 100 having a

2   communication network 101 that implements the opportunity tracking information system

3   (referenced interchangeably as "OTIS") for the automatic management of process flow

4   performance in accordance with the present invention. In the example of FIG. 1, one or more

5   user stations 102 (used interchangeably herein with "client computers 102," "workstations

6   102," and "clients 102") communicate over network 101 with at least one server computer

7   103 (used interchangeably with "server 103"). It will be appreciated by those skilled in the art

8   that the number of clients 102 and servers 103 may vary based upon design or user

9   requirements.

10   One embodiment of the network 101 in accordance with the present invention

11   includes the Internet. However, it will be appreciated by those skilled in the art that the

12   present invention works suitably well with a wide variety of computer networks over

13   numerous topologies, so long as network 101 connects the distributed user stations 102 to

14   server 103. For example, other public or private communication networks that can be used for

15   network 101 include Local Area Networks (LANs), Wide Area Networks (WANs), intranets,

16   virtual private networks (VPNs), and wireless networks (i.e., with the appropriate wireless

17   interfaces as known in the industry substituted for hard-wired communication links).

18   Generally, these types of communication networks can in turn be communicatively coupled to

19   other networks comprising storage devices, server computers, databases, and client computers

20   that are communicatively coupled to other computers and storage devices.

21   Each user works with system 100 to seamlessly access server 103 through a

22   workstation 102 interfaced to network 101. Referring now to FIG. 29, a first embodiment for

23   the client computer 102 is shown. The client computer 102 comprises a control unit 1350

-12-

1 coupled to a display device 1300, a keyboard 1322, a cursor controller 1323, a network

2 controller 1324 and an I/O device 1325 by a bus 1301.

3 Control unit 1350 may comprise an arithmetic logic unit, a microprocessor, a general

4 purpose computer, a personal digital assistant or some other information appliance equipped

5 to provide electronic display signals to display device 1300. In one embodiment, control unit

6 1350 comprises a general purpose computer having a graphical user interface, which may be

7 generated by, for example, a program written in Java running on top of an operating system

8 like WINDOWS® or UNIX® based operating systems. In one embodiment, one or more

9 application programs executed by control unit 1350 including, without limitation, word

10 processing applications, electronic mail applications, spreadsheet applications, database

11 applications and web browser applications generate the displays, store information, retrieve

12 information as part of OTIS 100. The control unit 1350 also has other conventional

13 connections to other systems such as a network for distribution of files (media objects) using

14 standard network protocols such as TCP/IP, http, and SMTP as will be understood to those

15 skilled in the art and shown in detail in Figure 29.

16 As shown in Figure 29, the control unit 1350 includes a processor 1302, main memory

17 1304, and data storage device 1307, all of which are communicatively coupled to system bus

18 1301.

19 Processor 1302 processes data signals and may comprise various computing

20 architectures including a complex instruction set computer (CISC) architecture, a reduced

21 instruction set computer (RISC) architecture, or an architecture implementing a combination

22 of instruction sets. Although only a single processor is shown in Figure 3, multiple processors

23 may be included.

-13-

1   Main memory 1304 may store instructions and/or data that may be executed by

2   processor 1302. The instructions and/or data may comprise code for performing any and/or

3   all of the techniques described herein. Main memory 1304 may be a dynamic random access

4   memory (DRAM) device, a static random access memory (SRAM) device, or some other

5   memory device known in the art. The memory 1304 preferably includes a web browser 1330

6   is of a conventional type that provides access to the Internet and processes HTML, XML or

7   other mark up language to generated images on the display device 1300. For example, the

8   web browser 1330 could be Netscape Navigator or Microsoft Internet Explorer. The main

9   memory 1304 preferably also includes a client program as will be described in detail below to

10   enable communication between the client computer 102 and the server 103 for creating,

11   editing, moving, adding, removing or viewing tasks.

12   Data storage device 1307 stores data and instructions for processor 1302 and may

13   comprise one or more devices including a hard disk drive, a floppy disk drive, a CD-ROM

14   device, a DVD-ROM device, a DVD-RAM device, a DVD-RW device, a flash memory

15   device, or some other mass storage device known in the art.

16   System bus 1301 represents a shared bus for communicating information and data

17   throughout control unit 1350. System bus 1301 may represent one or more buses including an

18   industry standard architecture (ISA) bus, a peripheral component interconnect (PCI) bus, a

19   universal serial bus (USB), or some other bus known in the art to provide similar functionality.

20   Additional components coupled to control unit 1350 through system bus 1301 include

21   display device 1300, keyboard 1322, cursor control device 1323, network controller 1324 and

22   audio device 1325. Display device 1300 represents any device equipped to display electronic

23   images and data as described herein. Display device 1300 may be a cathode ray tube (CRT),

-14-

1   liquid crystal display (LCD), or any other similarly equipped display device, screen, or

2   monitor.

3   Keyboard 1322 represents an alphanumeric input device coupled to control unit 1350

4   to communicate information and command selections to processor 1302. Cursor control 1323

5   represents a user input device equipped to communicate positional data as well as command

6   selections to processor 1302. Cursor control 1323 may include a mouse, a trackball, a stylus, a

7   pen, a touch screen, cursor direction keys, or other mechanisms to cause movement of a

8   cursor. Network controller 1324 links control unit 1350 to a network that may include

9   multiple processing systems. The network of processing systems may comprise a local area

10   network (LAN), a wide area network (WAN) (e.g., the Internet), and/or any other

11   interconnected data path across which multiple devices may communicate.

12   One or more I/O devices 1325 are coupled to the system bus 1301. For example, the

13   I/O device 1325 may be an audio device equipped to receive audio input and transmit audio

14   output. Audio input may be received through various devices including a microphone within

15   audio device 1325 and network controller 1324. Similarly, audio output may originate from

16   various devices including processor 1302 and network controller 1324. In one embodiment,

17   audio device 1325 is a general purpose; audio add-in/expansion card designed for use within a

18   general purpose computer system. Optionally, audio device 1325 may contain one or more

19   analog-to-digital or digital-to-analog converters, and/or one or more digital signal processors

20   to facilitate audio processing.

21   It should be apparent to one skilled in the art that control unit 1350 may include more

22   or less components than those shown in Figure 29 without departing from the spirit and scope

23   of the present invention. For example, control unit 1350 may include additional memory,

-15-

1   such as, for example, a first or second level cache, or one or more application specific

2   integrated circuits (ASICs). Similarly, additional components may be coupled to control unit

3   1350 including, for example, image scanning devices, digital still or video cameras, or other

4   devices that may or may not be equipped to capture and/or download electronic data to control

5   unit 1350.

6   Referring now to FIGS. 29 and 30, the server 103 will be described in more detail.

7   The server 103 preferably includes a control unit 1350 coupled to a display device 1300, a

8   keyboard 1322, a cursor controller 1323, a network controller 1324 and an I/O device 1325 by

9   a bus 1301. As shown in Figure 29, the control unit 1350 includes a processor 1302, main

10 memory 1304, and data storage device 1307, all of which are communicatively coupled to

11 system bus 1301. For convenience and ease of understanding like reference numerals have

12 been used for similar components used in both the client computer 102 and the server 103. In

13 the preferred embodiment, the server 103 is a multiple processor system such as a Dell 1800

14 made and sold by the Dell Computer Corporation, and the data storage device 1307 stores a

15 database.

16   In accordance with the present invention, OTIS 100 uses network 101 as a backbone

17 to allow component parts of an organization to communicate with database 104 and each

18 other. As shown in the example of FIG. 1, server 103 incorporates database 104 therein (i.e.,

19 locally), although other configurations of the database 104 and server 103 are well suited to

20 work with the present invention. For example, multiple remote databases can be

21 communicatively coupled to server 103 and throughout network 101.

22   Network 101 enables the communication between multiple components of server 103

23 and other devices, which may or may not be co-located, but may be distributed for

-16-

1 convenience, security or other reasons. To facilitate the communication between clients 102

2 and server 103, a client-server computer network operating system (NOS), which is an

3 operating system used to manage network resources, is used in conjunction with the present

4 invention. NOS can manage multiple inputs or requests concurrently and may provide the

5 security necessary in a multi-user environment. An example of NOS includes Windows NT

6 manufactured by the Microsoft Corporation of Redmond, Washington. Other operating

7 systems that are applicable include Windows 2000, Unix, Sun Microsystems' Solaris, and

8 Novell Netware.

9   Client 102 and server 103 of system 100 may beneficially utilize the present invention,

10 and may contain an embodiment of the process steps and modules of the present invention in

11 the form of a computer program. Alternatively, the process steps and modules of the present

12 invention could be embodied in firmware or hardware, and when embodied in software, could

13 be downloaded to reside on and be operated from different platforms used by real time

14 network operating systems.

15   Referring now to Figure 30, the memory unit 1304 is shown in more detail. In

16 particular, the portions of the memory 1304 needed for the processes of the present invention

17 are shown and will now be described more specifically. As shown in Figure 30, the memory

18 unit 1304 preferably comprises an operating system 1402, other applications 1404, at least one

19 OTIS application 1408, a first module 1414, a second module 1418, a third module 1410 an a

20 internet browser 1330. As noted above, the memory unit 1304 stores instructions and/or data

21 that may be executed by processing unit 1302. The instructions and/or data may comprise

22 code for performing any and/or all of the techniques described herein. These modules 1402-

23 418 are coupled by bus 1301 to the processing unit 1302 for communication and cooperation

-17-

1   to provide the functionality of the system 100. Those skilled in the art will recognize that

2   while the present invention will now be described as modules or portions of the memory unit

3   1304 of a computer system, the modules or portions may also be stored in other media such as

4   permanent data storage and may be distributed across a network having a plurality of different

5   computers such as in a client/server environment.

6   The operating system 1402 is preferably one of a conventional type such as,

7   WINDOWS®, SOLARIS® or LINUX® based operating systems.

8   The memory unit 1304 may also include one or more other application programs 1404

9   including, without limitation, word processing applications, electronic mail applications, and

10   spreadsheet applications.

11   The OTIS application 1408 is a procedure or routines that control the processor 1302.

12   Although only a single OTIS application 1408 is shown in the memory 1304 of Figure 30 for

13   ease of understanding the present invention, the server 103 will typically have several such

14   OTIS applications 1408; each application used for displaying information for a particular

15   organization. The OTIS application typically includes three layers of software according to

16   the present invention. These three layers are described below as the first, second and third

17   modules 1414, 1418 and 1410.

18   The Internet browser 1330 is of a conventional type and shown here for consistency

19   with Figure 29.

20   In one embodiment, the system 100 of the present invention includes a first module

21   1414, a second module 1418 and a third module 1410. The first module 1414 is a program for

22   controlling the server 103 and its components to enable communication across the network

23   101 with the client computers 102. The second module 1418 is a plurality of business objects

-18-

1   that interact with each other and the first and third modules 1414, 1410 to implement

2   processes for the functionality of creating, editing, moving, adding, removing or viewing tasks

3   consistent with the privileges assigned by position and user type. The third module 1410

4   controls the communication between the second module 1418 and the database 104. The third

5   module 1410 includes processes for storing data in and retrieving data from the database 104

6   as well as communication with the second module 1418. Exemplary functions and

7   implementation for the first, second and third module 1414, 1418, 1410 are described in more

8   detail below.

9   A particular embodiment, provided only by way of example, implements the system

10   100 as a three-module or three-tier Intranet application implemented with the Microsoft

11   Development Environment. The three modules or tiers 1414, 1418, 1410 include: (1) a first

12   module 1414 or front-end tier having server 103 embodied as an Internet Information Server

13   (IIS); (2) a second module 1418 or middle tier comprising Component Object Model (COM)

14   objects; and (3) third module 1410 or backend database tier (e.g., SQL). Those skilled in the

15   art will recognize that these particulars are provided only by way of example, and that a

16   variety of other implementations with other types software are within the scope of the present

17   invention.

18   Under the control of the first module, the server 103 communicates with the client

19   computer 102. For example, the first module may be an IIS, which is the software module

20   component of ActiveX that operates in a runtime environment enabling Active Server Pages

21   (ASPs) to interface therewith. ASPs generally provide a framework for constructing Web

22   applications using HTML, scripts and ActiveX components. The ASP page is created by

23   embedding such scripts within the HTML page. As a user makes the request for an ASP page,

-19-

1   the server 103 executes the scripts that have been embedded within the page so that the output

2   generated from the running the scripts is included in the HTML, thus allowing a client

3   browser on client 102 to view the page. It is noted that the present invention is well suited to

4   work with other formats for creating forms and processing input, including Dynamic HTML

5   (DHTML) technology. It will be evident to those skilled in the art that the client 102 is

6   adapted to run various types of commercially available browsers (e.g., Netscape, Internet

7   Explorer) suited to enable DHTML functionality. Furthermore, here and throughout this

8   application, the description of the present invention in the context of the Internet, browsers,

9   ASP, etc is only by way of example. Those skilled in the art will realize that the present

10   invention could be implemented on a variety of other hardware environments such as peer-to-

11   peer networks and mainframe systems just by way of example.

12   The second module or middle tier includes software to implement the business rules

13   and data access functionality corresponding to the organizational chart of the entity. The

14   second module is preferably implemented or encapsulated within one or more Component

15   Object Model (COM) objects. The COM objects are a way for software components to

16   communicate with each other. The COM objects are a binary and network standard that

17   allows any two components to communicate regardless of what machine they're running on

18   (as long as the machines are connected), what operating systems the machines are running (as

19   long as it supports COM), and what language the components are written in. COM objects

20   further provides location transparency: it doesn't matter to you when you write your

21   components whether the other components are in-process DLLs, local EXEs, or components

22   located on some other machine.

-20-

1   The present invention preferably uses COM objects to enforce rules about

2   transactions. For example, if a business rule is that you want no addresses in your database

3   that do not have valid zip codes then calling a COM object to validate the address may be

4   used. In this case it might check that the zip code was valid for the city and state specified.

5   Using a COM object allows many applications to share the same code. A COM object might

6   be used to validate the due date for a task e.g. to verify that it is in the future and that it falls on

7   a workday rather than a weekend or a holiday. The COM object might also check that the due

8   date does not fall on scheduled vacation day for the user to whom it is assigned. The present

9   invention includes a variety of COM objects to implement the functionality described below

10   with reference to the process flowcharts and screen shots of FIGS. 2-28. The versatility of this

11   approach lies in the coding of COM components, and the referencing of the COM

12   components from within the ASP pages. Such COM object can reside on the server 103 used

13   to serve up the objects. In the preferred embodiment, the second module 1418 or middle tier

14   layer of OTIS 100 is used to enforce a variety of business rules. For example, the second

15   module 1418 enforces access rights and viewing privileges. For users, the second module

16   1418 or middle tier restricts the display of tasks to only those tasks assigned to a particular

17   user. For managers, the second module 1418 or middle tier restricts the display and

18   performance of other operations on tasks to the tasks for the manager and those users assigned

19   to report to the manager. Finally, the second module 1418 or middle tier allow an

20   administrator to establish roles, disciplines and projects to which a user or position is assigned.

21   When a task is assigned to a position within the organization middle tier objects assure that

22   only individuals assigned to that position can access it. A final example of middle tier object

23   functionality within OTIS 100 is the use of COM objects to report exceptions within OTIS

-21-

1 100. This reporting may be as simple as changing the status of a task that is past due or more

2 complex such as sending an email at the time a task falls more than a predetermined time

3 behind schedule. Again, while the present invention is described in the context of COM

4 objects, it is only by way of example. Those skilled in the art will realize that the present

5 invention could be implemented using other software constructs, code and methods.

6 The third module 1410 or backend tier facilitates communication between server 103

7 and database 104. In this described embodiment, database 104 is preferably a relational

8 database accessed with Structured Query Language (SQL). It is noted though, that the present

9 invention is SQL independent; accordingly, those of ordinary skill in the art will recognize that

10 there is no specific requirement that a SQL database must be utilized. The functionality of

11 database 104 in accordance with the present invention is now discussed. In general, database

12 104 defines the details of the operation of system 100, and stores the information that is used

13 in accordance with the present invention. In particular and as will be described in greater

14 detail later, database 104 includes status information which defines a task (e.g., work item),

15 associated attributes according to certain business rules, and the organizational structure of a

16 business using the present invention.

17

18 OTIS Login

19 Those skilled in the art should recognize that the present invention will now be

20 described with reference to FIGS. 2-28, and while described in the context of a client-server

21 interaction across the Internet with the particulars of such communication, the present

22 invention is applicable to other contexts of communications between multiple users such as

23 users of a main frame computer, users of other proprietary network systems, and the

-22-

1   description here of the present invention in this specific context is only by way of example. It

2   should be understood that the processes and methods of the present invention are applicable to

3   any database being accessed by multiple users.

4   The operation of system 100 in accordance with the present invention is described

5   below, starting with reference being made to the flowchart of FIG. 2, which illustrates a

6   method by which a user gains access to OTIS. Once system 100 is running, the client 102

7   prompts the user for information to verify and authenticate the identification of the user.

8   One aspect of the present invention includes a security system that restricts access to

9   the system 100. With the security system, each user is associated with a user name (e.g., User

10 ID) and password to login, access and use the OTIS system 100. The login also creates a user

11 session, which will be discussed later in detail with respect to the level of access a user is

12 given in the nature of defined rights that can restrict the user's access to certain areas of the

13 system 100. For example, FIG. 6 shows one embodiment of a user interface (e.g., login

14 screen) 600 allowing the user to input the User ID 602 and password 604. It is noted that user

15 interface 600 includes a form 601, which is presented to the user at the client 102. Once that

16 information is input and submitted 201 (e.g., the user clicks on the OK button 606), the client

17 browser sends an instruction (e.g., HTTP request), including the User Id and password

18 entered, to server 103. In response to the HTTP request being received by server 103, a

19 session is invoked on server 103. As part of the session, the User ID 602 and password 604

20 are extracted from the HTTP request and compared 202 with a list of already established User

21 Ids and passwords stored on the database 104. If the combination of input information is

22 valid, then access to OTIS 100 is allowed 204 because the user information has been

23 authenticated. However, if the combination is invalid (i.e., NO branch of 202), then an error

-23-

1   message is 203 sent back from server 103 for display on client 102, after which the original

2   login screen 600 is re-displayed.

3

4   <u>OTIS Home Page</u>

5   Reference is made to the flowchart of FIG. 3 to describe the process of determining

6   whether a user has previously established a customized a display format or home page once

7   the user has successfully logged into OTIS 100. A home page is the user's view of the data

8   (i.e., the status information, which will be described subsequently in detail) in the system 100.

9   Server 103 invokes a process to obtain 301 a home page for the user. To make this

10   determination 302, server 103 accesses the database 104 to inquire about and retrieve the

11   customized home page associated with the state values (e.g., cookie values) received in the

12   request of for information (HTTP commands). If the user has not previously customized a

13   home page (e.g., NO branch of 302), then a default home page is retrieved 303. The default

14   home page is essentially a template with which status information can be incorporated. Next

15   in step 304, using the state values (e.g., cookie values) that identify the user, the server 103

16   queries the database 104 to locate and retrieve previously saved status information (e.g., tasks

17   assigned, anomalies). The status information is then incorporated with the default home page

18   (e.g., in DHTML form) and transmitted from server 103 for display 306 on client 102. For

19   example, FIG. 7 illustrates one embodiment of a user interface having a default home page

20   700 combined with the status information 702 retrieved from database 104. If status

21   information was not found on the database 104 for the particular user, then the default home

22   page is displayed 306 without data.

-24-

1   Still referring to FIG. 3, in step 302, if the user has a customized home page, then

2   server 103 retrieves it from database 104 as indicated in step 307. Thereafter, a determination

3   is made 308 as to whether the home page should be incorporated with status information. For

4   example, server 103 uses the state information from the client 102 (e.g., cookie values) to

5   locate ("personalized") status information saved on database 104 that pertains to the user. If

6   the personalized data does not exist, then the customized home page is formatted (e.g.,

7   DHTML) and transmitted from server 103 to client 102 for display 310. However, if

8   personalized status information does exist in step 308 (i.e., YES branch), then server 103

9   invokes a query on the database 104 to determine 311 the type of personalized status

10   information to retrieve. The type of information to be retrieved from database 104 can be, for

11   example, tasks 312, anomalies 313, or a combination 314 of tasks and anomalies. The

12   personalized status information retrieved is then incorporated with the customized home page

13   (e.g., in DHTML form) and transmitted from server 103 for display 315 on client 102.

14   Referring back to FIG. 7, once the user gains access to OTIS 100, the home page 700

15   can be customized with user preferred settings. In order to personalize the home page 700, the

16   user clicks on the top bar 704 of the table referenced as "Customize View." This in turn

17   presents to the user the option to view the current table, to change the sort order or to change

18   the number of hours ahead that the table indicates. Once the user is finished customizing the

19   views, the user preferences are stored by server 103 in database 104 using the state

20   information (cookies) associated with the user logged into the user session. This aspect of the

21   present invention pertaining to saving a customized home page enables users the ability to

22   return to the previous settings associated with the former login session.

23

1　The Operation of OTIS

2　　　A. Definition of Terms

3　　　Before discussing the architectural design, requirements, business rules and

4　development environment of the present invention, several definitions are introduced to clarify

5　the terms used herein. A task is defined to mean any work item that pertains to anything that

6　requires attention. That is, a task is a work item that needs to be accomplished. An anomaly

7　is defined to mean any development error in a project requiring attention. A position is

8　defined to mean a person of the organization using their title as their reference (e.g., a

9　developer). A position class is defined to mean a group of positions.

10　　　B. General Overview and Classification of Individuals Using OTIS

11　　　One aspect of the present invention includes a management tool that allows multiple

12　views of what is to be done within an organization. That is, OTIS 100 allows individuals to

13　view information about: the tasks to be accomplished and the anomalies to be rectified.

14　Additionally, OTIS 100 allows individuals to update the status of the work items and

15　anomalies, and to enter new work items and anomalies. The individual's class and position on

16　the organization chart dictate the extent to which an individual may manipulate the attributes

17　of a work item. Such an exemplary organizational chart with the class and position for

18　different individuals in the organization is shown in FIG. 31

19　　　Access to certain functions and features of the present invention require certain rights

20　afforded to individuals using OTIS 100. Such rights are based upon a classification of the

21　type of individuals using OTIS 100. In general, classes include users who have basic rights;

22　managers who have the rights to assign tasks to people who report to them in the organization

23　chart; and administrators who can modify control parameters in the system 100. One type of

-26-

1 individual is a user, and a user is defined to mean any individual accessing and using OTIS

2 100. For example, within a business, a user is typically an employee. Referring to FIG. 31,

3 positions 7-12 in the organizational chart are users. A manager is another type of individual,

4 and is defined to be a supervisor of the organization, which may be a company, business,

5 institution, or non-profit entity. Managers using OTIS are provided the right to maintain and

6 assign (or reassign) tasks to be completed by individuals (e.g., users) whom they supervise

7 within an organization infrastructure. Referring to FIG. 31, positions 1-5 in the organizational

8 chart are managers. It will be apparent that managers (e.g., positions 2-5) in turn can be

9 supervised by higher-level managers (e.g., position 1). Another type of individual is the

10 administrator, which is defined to mean those having the responsibility to implement and

11 maintain the organization or site. The rights of administrators allow them complete access to

12 all functions and features of OTIS 100 (e.g., to do anything in the system). Referring to FIG.

13 31, positions 5 and 6 in the organizational chart are administrators. In one embodiment, these

14 rights afforded based on the classification of the individual using OTIS 100 can be

15 implemented by a "system administrator" of OTIS 100.

16 OTIS includes the management structure of the organization within database 104. For

17 example, the "organization chart" is entered interactively, using standard industry practices

18 such as "drag and drop" to allow the modification of the organization chart which is

19 represented graphically. Lines of authority and responsibility are indicated in this database

20 104.

21 OTIS is designed to track anomalies, routine automatic tasks, and manually inputted

22 tasks by the positions of those individuals as defined within an organization. OTIS stores the

23 status information about work items to be accomplished in the database 104. Work items

-27-

1   have multiple attributes associated with them. The attributes include, but are not limited to,

2   description, required completion date, priority, duration, originator and assignee. The

3   attributes are entered into the system interactively via the communication network 101. There

4   are a variety of ways to organize the tasks. For example, work-items can be grouped by

5   projects. To associate a project with a work item, each work item is uniquely identified by a

6   serial number assigned to it at the time it is created, and generally, this number may not be

7   changed. The changing or setting of the attributes is controlled so that only authorized

8   individuals may create and/or modify them. Work items may occur only once or may be

9   repetitive.

10   Once a work-item has been created it must be assigned to a responsible individual (or

11   group of individuals). The assignment may only be made by an individual's manager. OTIS

12   tracks the due date and completion of each item. It allows users to enter the fact that they have

13   completed tasks or anomalies, and managers to verify the completion of the tasks or

14   anomalies. Tasks that are not completed on time may be reported to the manager of the

15   assignee as determined by the organization chart. The time delay between the failure to

16   complete and the notification is a system parameter. The action of notifying higher levels of

17   management is termed escalation. Management at even higher levels may also be notified at

18   the same time or after a further delay. The notification takes place over the communication

19   network.

20   OTIS allows all classes of individuals to view the state of work-items within the

21   system. Each user may customize how they view the work items. All users may control

22   which items they see. They may be selected by status and sorted by multiple criteria. The

23   ability to see an item at all is controlled by project; users are assigned to projects and may only

1 view items on that project. Additionally, items may be made visible or invisible to outside

2 auditors (for example customers) who may view selected items in their projects.

3   The functionality described above is integrated into a system 100 that automatically

4 uses position and class to determine what operations a user is able to perform. An exemplary

5 method for operation of the invention is shown in FIG. 32. The process begins by receiving

6 3202 a request for access to the system 100, and authentication 3204 of the user. Then based

7 on the input user information, the position of the user is determined 3206. The position of the

8 user determines whether they are a user, manager, or administrator, and what other users or

9 positions report to them. Then the tasks associated with the determined position are retrieved

10 3210. This could include task of other positions that report to the determined position in the

11 organizational chart. Next, the process determines 3212 whether the user can create, remove

12 or edit a task. If so the method confirms 3214 that the position is allowed to perform the

13 operation before performing the operation 3216 on the task and saving the updated data to the

14 database. After either step 3212 or step 3216, the method determines 3218 whether the user is

15 assigning a task. If so the method verifies 3220 that the position is a manger and that the

16 position is allowed to perform the operation before assigning 3222 the task and saving the

17 updated data to the database. After either step 3218 or step 3222, the method determines 3224

18 whether the user is attempting to change a position, class, project membership or some other

19 administrative operation. If so the method verifies 3226 that the position is an administrator

20 and that the position is allowed to perform the operation before executing the operation 3228

21 on the task and saving the updated data to the database. After either step 3224 or step 3228

22 the process is complete and ends. As can be seen from this process, the modifications to the

23 process flow are high and seamlessly integrated into positions in the organizational chart.

C. Anomaly Overview

One aspect of the present invention is to assist supervisors with tracking the progress of a project. One example for doing so is with the present invention tracking anomalies. With all of the anomalies listed in one central location, it is easier for the developer as well as the supervisor to review the project status because the outstanding issues, i.e., development error in a project requiring attention, can be immediately discerned and focused upon. One aspect of the security feature in OTIS 100 enables the developers to "see" only the project(s) they are assigned to.

One technical advantage of the present invention is that anomalies are designed to track the problems associated with any kind (i.e., subject matter) of project. Once a user accesses OTIS 100, the user can enter, review, and edit an anomaly subject to those rights afforded by the administrator, which for example may be on a project-by-project basis.

Reference is made to FIG. 8, which illustrates an example of a user interface 800 enabling a user to describe in detail the anomaly and the anomaly type. In general, an individual with the user right can view this interface to supply input status information to the data input screen, that is, to assign and define the anomaly for a project. In one embodiment in accordance with the present invention, identifying information can be input to describe the anomaly as follows. For example, a project indicator 802 identifies the project that has the anomaly, and the user can select the particular project from a list of projects that the user has been assigned to. A discipline indicator 804 identifies the corresponding discipline that applies to the selected project. An application (App) version indicator 806 designates the version of the current application. A description field 808 allows the user to enter a detailed

-30-

1   description of the anomaly. Using a bug type field 810, the user can indicate a particular bug

2   associated with the anomaly by selecting such from a predefined list. The user inputting the

3   information about the anomaly can be identified by the Input By field 812. An Input Date

4   field 814 allows the user to indicate the date and the time in which the anomaly was input. A

5   control number field 816 indicates an individual number assigned to the anomaly for tracking

6   purposes, and preferably should not be changed.

7   One advantage of having the above-information inputted is to let the project developer

8   know what needs to be fixed or finished and by what time. The task part of OTIS 100 will

9   allow supervisors to assign reoccurring tasks to a position (individual). If the individual

10   assigned is unavailable, another person may be substituted and by reviewing this information,

11   will know all of the daily tasks that need to be completed.

12   <u>D. Assignment information</u>

13   Another aspect of the present invention gives managers the ability to assign an

14   individual (i.e., typically a user, but sometimes another manager, usually a subordinate) the

15   task of fixing the anomaly. Reference is made to FIG. 9 to illustrate an example of a user

16   interface 900 enabling a manager to assign a priority level 902 to the anomaly. The priority

17   level indicates the urgency of having the anomaly rectified, and provides an indication of

18   which anomalies should be given immediate attention to as opposed to those anomalies that

19   can be addressed subsequently. For example, a priority of 1 through 9 can be assigned in field

20   902, where 1 represents the highest priority and 9 represents the lowest priority. With the

21   Assigned To field 904, the manager can designate the individual who will assume

22   responsibility for rectifying the anomaly. A drop-down menu is provided to facilitate the

23   selection of the individual assigned the task, however, the individual generally should have

1   been previously assigned rights to work on the particular project. The manager can also

2   indicate a date in field 906 for which the anomaly is estimated to be fixed by, as well as an

3   estimate of the number of hours in field 908 to resolve the anomaly. In field 910, the manager

4   indicates a method, which represents whether the customer is permitted to see the anomaly or

5   not.

6

7   E. Fixed information

8   In response to being assigned a task and completing the task, including rectifying an

9   anomaly, users can provide notification to OTIS 100 of the completion of the task. Generally,

10   individuals having user rights can only mark off their work items, while by comparison;

11   managers can edit all entries with information pertaining to the completion of a task.

12   For example, reference is made to FIG. 10 to illustrate an example of a user interface

13   1000 enabling: (1) a user assigned to fix an anomaly the ability to mark off the fact that the

14   anomaly has been fixed; and (2) a manager the ability to complete the process by approving

15   (e.g., completely signing off) the completion of the anomaly. In the example shown, the new

16   version number of the project can be entered in the field entitled Fixed Version 1002. The

17   individual resolving the anomaly for the project can be identified in the Fixed By field 1004.

18   In the field labeled Fixed Date 1006, the individual enters the date on which the anomaly was

19   fixed. These entries for these described fields in general should be filled in by users.

20   Still referring to FIG. 10, for the following fields, it is preferable that managers be

21   given access to enter data. The QA Push Date field 1008 indicates date for which the new

22   version of the project was moved to Quality Assurance (QA) or to Production. The Prod Push

23   Date field 1010 indicates the date that the new version of the project was pushed to the

-32-

1   production system. The Signed Off By field 1012 indicates the identity of the manager who

2   signed off that the anomaly has been fixed. A drop-down menu can be provided to list such

3   managers having the authority to approve the completion of the particular project. The Signed

4   Off Date field 1014 indicates the date that the manager signed off the completion of the

5   anomaly.

6   Users, managers and administrators generally receive information and interact with

7   OTIS 100 in several ways. One manner in accordance with the present invention comprises

8   interactively receiving information and interacting with OTIS 100 via a web browser running

9   on the workstation (i.e., client 102). The browser displays forms on the workstation

10   presenting information from OTIS 100 and allows, where appropriate, for the user to enter

11   information into the system 100. The data that can be reviewed at the client 102 is dependent

12   upon the privileges and position of the user within the organization.

13   A second manner in accordance with the present invention comprises individuals

14   obtaining information from the system 100 by electronic mail. Parts of OTIS 100 allow for

15   email notification, wherein email messages can be formatted and sent based on the content of

16   database 104. For example, email messages can be sent to users to inform them of work items

17   that are scheduled to be accomplished. Further, email messages can be sent to inform

18   managers of work-items that have not been accomplished on schedule (i.e., in a timely

19   manner). The parameters associated with these communications are preferably stored in

20   database 104

21

1 F. Screen Capture

2 Generally, individuals with any type of right (e.g., users, managers, administrators)

3 can view and use the screen capture feature. For example, a user can capture the error being

4 viewed on screen by inputting a command such as ALT+PrintScreen at client 102. The user

5 can then navigate to the input screen to save the image. For example, a "Get Image" button

6 can be provided, which upon invocation takes the screen capture from the clipboard and

7 through an email module (e.g., NetTransport™) sends the image to the third module were it is

8 stored in the database 104. This helps in finding the exact error. In an alternative

9 embodiment, an individual can attach a file or multiple files to the anomaly and store it in the

10 database 104. In an alternate embodiment, a button that brings up a help screen to guide the

11 user through these steps can also be provided.

12 G. Reviewing Anomalies

13 In one aspect of the present invention, managers can view any anomaly through a

14 novel display format. Referring to FIG. 11, an illustration is shown of an example of a user

15 interface 1100 (review screen) for displaying to a manager for review any anomaly. In the

16 example shown, the review screen illustrates that the anomalies can be viewed in a tabular

17 format of row and columns similar to a spreadsheet. Additionally, the priority assigned to an

18 anomaly determines what color the row is so the individual (e.g., manager) may quickly see

19 the priority of the anomalies. The anomalies can be listed by the project 1102 they are

20 assigned to, a user specified or unassigned 1104, bug types 1106 and input users 1108. By

21 contrast, those users who may view anomalies are generally assigned to the particular project

22 associated with the anomaly. For example, a user can review the pending tasks and anomalies

-34-

1   as shown in FIG. 12. It is noted that a manager would see more details of a task and anomaly,

2   similar to the details shown in FIG. 12, by clicking upon one of the entries in FIG. 11.

3   Whichever level of access an individual (i.e., user v. manager) has to review the list of

4   anomalies, the individual can sort entries by a variety of parameters. For example, such

5   parameters include the control number, project name, version, bug type, priority, assigned to

6   estimated fix date, signed off date, and input user. Individuals can also "turn off" rows that

7   contain signed off data, past due tasks, and anomalies that are not done. They may also click

8   on the control number to take them back to the input screen to edit the anomaly. It is

9   preferable though, that no user, except the original inputting user, is allowed to edit the text in

10   the assignment information.

11   Referring now to FIG. 4, one example of implementing how status information (e.g.,

12   tasks and anomalies) is reviewed or edited depending upon the level of access that an

13   individual has is illustrated in the flowchart shown. In the example shown, the individual is

14   assumed to be of user type. When the user selects the Review Anomaly option (e.g., clicks on

15   button 1202 of FIG. 12), OTIS 100 detects 401 a selection made and passes control to step

16   402. The privilege of the user is checked 402. Access is disallowed 403 if the privileged does

17   not exist. One example of implementing this verification is for server 103 to invoke an SQL

18   query of database 104 to determine whether access has been previously granted to the feature

19   that the user is attempting to review. If sufficient privilege exists (e.g., Yes branch of 402),

20   then access is granted and the results of the status information are retrieved from database 104

21   and displayed in step 404. Subsequently, a determination is made 405 as to whether the user

22   wants to select the order in which the tasks and anomalies are to be reviewed or wants to

23   proceed directly to editing the task or anomaly. If the order is to be changed, the user selects

-35-

1   the order 406, and the information is reordered and then displayed in the selected order by

2   returning to step 404. It is noted that the selection of the order can be a repetitive process (as

3   shown) or a single control input selection, depending upon implementation choices. At step

4   405, if the user selects to edit the information displayed, the process continues in step 407. In

5   step 407 editing is input at the client 102 (e.g., by a mouse at the workstation) and the

6   information is sent over the network to the server 103. After making the necessary and

7   allowed changes, the update (e.g., edit) is either saved in the database 104 or cancelled 408

8   based on user input.

9

10   H. Tasks Overview

11   1. Data Entry

12   The aspect of the present invention concerning the tracking of tasks enables

13   supervisors to assign recurring tasks to a position class or a specific position. If an individual

14   assigned to a particular position is unavailable, another person may step into that position and

15   will be able to know all the tasks of that day that needs to be completed in an easy and

16   efficient manner using the present invention.

17   There are many types of tasks allowed. There are one-time tasks that will occur once

18   and then be done with. There are also repetitive tasks. These repetitive tasks can be broken

19   down into flexible or fixed. A task that is assigned as flexible will be allowed to move in time

20   (e.g., a holiday's weekends, days off, etc.). A fixed task will not be allowed to move. Once a

21   flexible or fixed indication is assigned to a task, the indication will stay assigned even if that

22   day happens to be a holiday; and preferably, the indication should not be reassigned.

-36-

1   In accordance with the present invention, tasks as defined for OTIS 100 allow a

2   person to assign tasks to a position. One example for implementing tasks includes the main

3   tasks being stored in a table and every day a background process or program will go through

4   and put all the current day's tasks in an auxiliary table. Once a task is completed it will stay in

5   this table and become the log entry.

6   Users may input tasks for themselves, while it is preferable that managers can assign

7   tasks to a position other than their position. In one embodiment in accordance with the present

8   invention, the positions that managers are allowed to assign tasks to comprise those positions

9   below them in the organization chart (e.g., to subordinates, like users).

10   Reference is made to FIG. 13, which illustrates an example of a user interface 1300

11   that allows the user to enter the information about the task. In the example shown, the user

12   can enter a detailed description 1302 of the task, and specific directions 1304 that are required

13   to complete the task. Other fields shown in FIG. 13 include: an indication of the position class

14   1306, which may selected from a list provided in a drop-down menu bar; a position 1308,

15   which the user may select from a list of all positions in the selected position class; and a task

16   name 1310, which allows the naming of tasks for easier and quick referencing. Additionally,

17   it is noted that the screen input shown in FIG. 13 is also available to a manager, and is a screen

18   entry where a task can be assigned to a position class or a specific position.

19   Additional examples of data entry in accordance with the present invention are now

20   mentioned below. In FIG. 14, an example is shown of a user interface 1400 for inputting

21   information used to track the date 1402 the task was inputted and the priority 1404 of the task.

22   Additional fields that are shown in the example include the user name 1406 that entered the

23   task and the project 1408 associated with the task.

1   In FIG. 15, an example is shown of a user interface 1500 for the manager to assign the

2   amount of time for a task once it is assigned to the position. Managers can determine if the

3   task needs to be done by a certain date 1502 or time 1504 if, for example, the task can be

4   resolved either in a one-time servicing or in a repetitive manner (e.g., a certain amount of

5   days/hours, especially when a task is repetitive). Field 1506 entitled Escalate When Not Done

6   permits the task to be escalated when it is not timely completed. The manager assigning the

7   task may also assign the repeating time interval so that the system will send notification to the

8   assignee. Additionally, once a task is escalated, the system 100 will notify the manager

9   associated with the task that has been escalated according to a repeating time interval. This

10   also allows for notification when the task is not completed.

11   FIG. 16 illustrates an example of a user interface 1600 where managers are allowed to

12   check when they want email notification. In the example shown, the appropriate manager can

13   receive email notification 1606 when the task is not completed on time 1602 or when the task

14   was escalated 1604 to the next position defined in the organization chart.

15   FIG. 17 illustrates an example of a user interface 1700 indicating 1702 the occurrence

16   of a task on the input screen as a sentence. In FIG. 18, a dialog box allows the user to assign

17   the occurrence of the task, for example, by selecting the month, week, day or hour that the task

18   will occur and how often it will occur. This dialog box can be invoked if a "Change" button is

19   clicked.

20   In FIG. 19, an example is shown of a user interface 1900 with a Review Tasks screen.

21   In the example shown, a user can view and mark off tasks. The user can also restrict how

22   many days ahead that tasks should be displayed for review. Additionally, the user is able to

-38-

1 sort the columns. For example, columns can be sorted by priority or "complete by time" field.

2 This screen is also used for managers to view all the master tasks and edit or delete them.

3 ### 2. Escalation

4 A user interface for escalating a task that remains incomplete can also be included in

5 accordance with the present invention. If the task requires escalation (i.e., determined by the

6 inputting manager), and the task was not completed by the specified time, the task is escalated

7 up to the next position as specified in the organization chart. Email notification can happen at

8 every step that the task is escalated up or the inputting manager can select notification only

9 when the task reaches a certain position. In either case, the email is sent to the positions

10 specified. If a user cannot complete a task, an explanation may be given in that particular task

11 log. This will mark it done (e.g., completed) by the system 100 and escalation will not occur.

12 Notification will be e-mailed to the position specified.

13

14 ### 3. Task Templates

15 Task templates allow a manager to group tasks together and create a template that can

16 be assigned to a new project. Each of the tasks within the template is assigned to the specified

17 position classes and specified occurrence per the requirements of the template. Once a

18 template is created and assigned, the manager can still go into the template and add a task or

19 anomaly at anytime.

20 When the template is created, the managers are able to use any of the tasks or multiple

21 tasks that are already in the system 100. The managers may also use any templates within

22 OTIS 100. The template will generally not use the information comprising the occurrence of

23 the task or whom the task is assigned to. The manager assigns this information separately for

-39-

1   the template. It is preferred that the manager assign the task to a position class, so the task

2   may be later assigned to different positions in the position class.

3   Referring to FIG. 20, an example is shown of a user interface 2000 for the manager to

4   assign the task, wherein a wizard 2002 will pop up. In the example shown, the manager

5   assigns the project and if one is not in the list, the manager may add a new project 2004. The

6   manager also assigns 2006 all the templates and tasks that are desired for this particular project

7   as well as the start date 2008 of the project.

8   In FIG. 21, an example is shown of a user interface 2100 (e.g., a second screen of the

9   wizard) that lists all the position classes 2102 in the template and positions 2104 within the

10 class that the tasks can be assigned to. Any position in the tasks will automatically be given

11 rights to the project.

12   FIG. 22 illustrates an example of a user interface 2200 that enables a manager to

13 review all the templates that exist for projects for which they are assigned. Additionally,

14 administrators can use interface 2200 to view all of the templates for the system. With the

15 example shown, these two types of individuals, managers and administrators are able to view

16 the templates by name or by clicking on a checkbox at the top to allow viewing of the tasks or

17 templates assigned to the template. The user interface 2200 also provides the manager with

18 the ability to edit or delete the template. If the template is deleted, only the template gets

19 deleted. None of the tasks in the template are deleted as the tasks are deleted separately.

20   From anywhere in OTIS 100, the individuals can click on the calendar option from the

21 menu and be taken to the calendar screen, which is shown in the user interface 2300 of FIG.

22 23. There they will be able to see their tasks on the calendar.

1   With the month view option, the individuals can see their tasks and the estimated fix

2   days laid out on a monthly calendar. In one embodiment, the days with an anomaly estimated

3   fix date being within the next two days are shown red. Also, any anomaly with a priority of

4   three or more is indicated in red on the month for clarity and focus. The user may go forward

5   and see their tasks as well as backwards.

6   In another embodiment in the nature of the weekly view, as seen in the user interface

7   2400 of FIG. 24, the user is also able to view anomalies and tasks on a weekly basis. In the

8   example shown in FIG. 24, the control number is displayed in the table and linked to the

9   anomaly or task. Clicking on the link will take the user to that task or anomaly.

10

I. General Administration

11

12   One aspect of the present invention is that OTIS 100 is embodied as a web site

13   accessible on an intranet, which is maintained through administrative features of the web site.

14   The administrative portions of the web site are accessible only to those individuals being

15   administrators, who can enter in new projects and assign users and disciplines to that project.

16   FIG. 25 shows an example of a user interface 2500, where administrators can enter this

17   information. All of the project names are maintained and new ones can be added. For each

18   project, the administrator preferably assigns the disciplines and the users allowed on that

19   project.

20   FIG. 26 illustrates an example of a user interface 2600 where the disciplines are

21   assigned to a project. In the example shown in FIG. 26, the discipline breaks down the

22   specific categories in the project. This helps narrow the location of the anomaly within the

23   project. The disciplines can be designed not to be global.

-41-

1   In FIG. 27, an example of a user interface 2700 is shown. In the example, the

2   administrator can add users to the project for them to view and add the anomalies, as well as

3   edit the anomalies. The administrator can add users that have rights and who are preprocessed

4   within the security system of OTIS 100.

5   FIG. 28 illustrates an example of a user interface 2800 listing all of the anomaly types

6   that are maintained and the new ones that are added. In the example shown, the anomaly

7   types preferably are global so they pertain to all the projects with OTIS 100.

8   Other types of administrative functions in accordance with the present invention will

9   now be discussed. For example, the position names are maintained and new ones are added

10   with one aspect of the present invention. Once a position is added, users in the security system

11   must be added to the positions. Generally, one user is assigned to a position, but a user may

12   be assigned to multiple positions. Once positions are created they are then placed in classes.

13   Multiple classes can share one position.

14   An additional administrative function concerns the organization chart. This chart

15   allows the administrator to define the structure of those positions within an organization like a

16   company. All positions can be shown in a tree format, and the administrator may move the

17   position around and reassign the parent of a position. The same position may also be assigned

18   to multiple parents.

19   Yet another administrative function comprises the Reports and logs. Those

20   individuals being of the type having manager and administrator rights can view the reports and

21   logs. For example, several user activities that are logged in accordance with the present

22   invention include: (1) login; (2) adding or editing of an anomaly or task; (3) marking or

-42-

1 signing off of an anomaly or task; (4) escalation of tasks; (5) creation, assignment, and editing

2 of templates; and (6) all administration functions.

3 Additionally, a manager can see in a tabular view of rows and columns like a

4 spreadsheet of those tasks that have been escalated and to whom. A manager can further view

5 all anomalies that have been assigned and are past the estimated fix date.

6 Reference is now made to FIG. 5, which illustrates a flowchart showing an example of

7 a process for providing status information concerning the administrative features, and for

8 incorporating updates thereto. At step 51, the type of administrative function is selected by the

9 user, whereupon server 103 queries database 104 to retrieve corresponding information so that

10 a display screen such as a web page can be formatted with the administrative status

11 information and sent to client 102 for display 52 thereon. Control then passes to step 53,

12 where a determination 53 is made as to whether there exists data input by an individual. If

13 there is no data, then a default screen such as the home page is displayed 54. Otherwise, if

14 there is data, then control passes to step 55. At step 55, a determination is made as to whether

15 the data is formatted correctly. If not, then an error message is displayed 57. However, if the

16 data is formatted correctly, then server 103 saves the data 56 in database 104. Thereafter, the

17 default screen such as home page is displayed 58 populated with the updated information.

18 While the invention has been described in conjunction with the described

19 embodiments, it is evident that many alternatives, modifications and variations will be

20 apparent to those skilled in the art in light of the foregoing description. Accordingly, the

21 disclosure of the present invention is intended to be illustrative, but not limiting, of the scope

22 of the invention, which is set forth in the following claims.